A universal image coding approach using sparse steered Mixture-of-Experts regression

Ruben Verhack, Thomas Sikora, Lieven Lange, Glenn Van Wallendael, and Peter Lambert

In: 2016 IEEE International Conference on Image Processing (ICIP), 2142–2146, 2016.

A UNIVERSAL IMAGE CODING APPROACH USING SPARSE STEERED MIXTURE-OF-EXPERTS REGRESSION

Ruben Verhack[⋆†]*, Thomas Sikora*[†]*, Lieven Lange*[†]*, Glenn Van Wallendael*[⋆]*, and Peter Lambert*[⋆]

[⋆]Ghent University - iMinds - Data Science Lab, Ghent, Belgium
[†]Technische Universität Berlin - Communication Systems Lab, Berlin, Germany

ABSTRACT

Our challenge is the design of a "universal" bit-efficient image compression approach. The prime goal is to allow reconstruction of images with high quality. In addition, we attempt to design the coder and decoder "universal", such that MPEG-7-like low-and mid-level descriptors are an integral part of the coded representation. To this end, we introduce a sparse Mixture-of-Experts regression approach for coding images in the pixel domain. The underlying stochastic process of the pixel amplitudes are modelled as a 3-dimensional and multi-modal Mixture-of-Gaussians with K modes. This closed form continuous analytical model is estimated using the Expectation-Maximization algorithm and describes segments of pixels by local 3-D Gaussian steering kernels with global support. As such, each component in the mixture of experts steers along the direction of highest correlation. The conditional density then serves as the regression function. Experiments show that a considerable compression gain is achievable compared to JPEG for low bitrates for a large class of images, while forming attractive low-level descriptors for the image, such as the local segmentation boundaries, direction of intensity flow and the distribution of these parameters over the image.

***Index Terms*—** Mixture of Experts, Gaussian Mixture Regression, Steering Regression, Gaussian Mixture Models, Sparse Image Coding

1. INTRODUCTION

Image and video compression has been a field of intense research over the last 30 years with tremendous impact on practical implications [1]. Our goal is the development of an efficient "universal" image coder, in which the format generates a bit-efficient coded bitstream with excellent image reconstruction quality – and easy bit-level access to MPEG-7-like low- and mid-level image features at the decoder [2]. These features are essential for various image processing tasks, e.g. classification, and image comparison.

Ideally, our compression strategy is designed to intrinsically support edge-preserving super-resolution enhancement or downsampling of the decoded images. This calls for space-continuous and parametric "edge-sensitive" sparse representations of an image – to allow description and redundancy reduction in the pixel domain rather than in the frequency domain. As such, we drastically depart from established block-based transform or wavelet domain image coding paradigms in JPEG and JPEG 2000.

Our approach is motivated by the work on *Steered Kernel Regression* (SKR) by Takeda et al [3], which produces excellent edge-preserving results for image denoising and super-resolution applications. In our own recent work, we adopted this SKR strategy for an image coder (SSKSC) that is based on irregular sub-sampling with SKR regression at the decoder side [4]. For coding, SKR has the particular shortcoming of having only local support. As such, the level of sparsity that can be achieved is too limited.

In this paper, we introduce a sparse *Steered Mixture-of-Experts* (SMoE) representation for images that provide local adaptability with global support. This representation drastically departs from SKR and SSKSC in that the kernels that are employed are global (hyper-)planes centered in irregular positions in the image domain. The SMoE representation is used directly to model the images for coding and not as a post-processing strategy. This approach is scalable in dimensions, e.g. spatio-temporal, in which motion vectors are made redundant as these are modelled by space-time correlation [5]. SMoE crosses the border with the field of machine learning, as it is closely related to Support Vector Regression [6], Radial Basis Functions Networks and Artificial Neural Networks. These techniques require more computational power than traditional DCT approaches [1]. However, due to the recent advances, these techniques have become more feasible.

Our coding philosophy is deeply embedded in a Bayesian framework. Our underlying assumption is that image pixels are instantiations of a non-linear or non-stationary random process that can be modelled by spatially piecewise stationary Gaussian processes. As such, the model takes into account different regions of the image, their segmentation borders and

The research activities described in this paper were funded by the Data Science Lab (Ghent University - iMinds), Communication Systems Lab (Technische Universität Berlin), the Agency for Innovation by Science and Technology in Flanders (IWT), the Fund for Scientific Research Flanders (FWO Flanders), and the European Union.

edges. We assume that the random process is modelled by a space-continuous *Gaussian Mixture Model* (GMM). The encoder modeling and analysis task thus involves estimating the parameters of the model. Since we allow the Gaussian probability distribution functions (pdf) to steer, we enable the desired steering regression capability. Each 3-D Gaussian component then acts as an "expert" in its respective arbitrary shaped image region. All experts collaborate in a Mixture-of-Experts framework [7], thus one parametric, continuous regression function for the entire image is derived. This SMoE has the tight structure of a parametric model, yet still retains the flexibility of a non-parametric method [8].

2. STEERED MIXTURE-OF-EXPERTS REGRESSION

2.1. Gaussian Mixture Regression

In general, the goal of regression is to optimally predict a realization of a random vector $Y \in \mathbb{R}^q$, based on a known random vector $X \in \mathbb{R}^p$. Note that the joint pdf $p_{XY}(x,y) \in \mathbb{R}^{p+q}$ contains all the necessary information that can be known about the random processes.

Gaussian Mixture Models (GMM) are frequently used to approximate multi-modal, multivariate distributions $p_{XY}(x,y)$. The parameters can be estimated from the training data by the *Expectation-Maximization* (EM) algorithm [9]. We arrive at *Gaussian Mixture Regression* (GMR) as follows [8]. Assume training data $D = \{x^i, y^i\}_{i=1}^N$ with joint probability density:

$$p_{XY}(X,Y) = \sum_{j=1}^{K} \pi_j \mathcal{N}(\mu_j, R_j) = \sum_{j=1}^{K} \pi_j \phi_j \qquad (1)$$

and $\sum_{j=1}^{K} \pi_j = 1, \mu_j = \begin{bmatrix} \mu_{X_j} \\ \mu_{Y_j} \end{bmatrix}, R_j = \begin{bmatrix} R_{X_jX_j} & R_{X_jY_j} \\ R_{Y_jX_j} & R_{Y_jY_j} \end{bmatrix}$

The parameters of this model are $\mathbf{\Theta} = [\Theta_1, \Theta_2, ..., \Theta_K]$, with $\Theta_j = (\pi_j, \mu_j, R_j)$, respectively being the population densities, centers and covariances. A normal pdf of dimension $p+q$ can be factorized as

$$\mathcal{N}_{p+q}\left(\begin{bmatrix} \mu_X \\ \mu_Y \end{bmatrix}, \sigma^2 \right) = \mathcal{N}_q(\mu_{Y|X}, \sigma_Y^2)\mathcal{N}_p(\mu_X, R_{XX})$$

and accordingly

$$p_{XY} = \sum_{j=1}^{K} \pi_j \phi_{Y|X_j}(m_j(x), \sigma_j^2)\phi_{X_j}(\mu_{X_j}, R_{X_jX,j}) \qquad (2)$$

$$m_j(x) = \mu_{Y_j} + R_{Y_jX_j} R_{X_jX_j}^{-1}(x - \mu_{X_j}) \qquad (3)$$

$$\sigma_j^2 = R_{Y_jY_j} - R_{Y_jX_j} R_{X_jX_j}^{-1} R_{X_jY_j} \qquad (4)$$

Notice, that $m_j(x)$ is a linear hyper-plane in $\mathbb{R}^{p+q}$ with a $(p+q)$-dimensional slope defined by $R_{Y_jX_j} \times R_{X_jX_j}^{-1}$, a desired linear steering kernel that provides global support over the entire signal domain.

A signal at location x is estimated by the weighted sum over all K mixture components (Eg. 7). Every mode in the mixture model is treated as an expert and the experts collaborate towards the definition of the regression function. Note that the reconstruction is smoothed *piecewise linear*. Early work on compression of piecewise smooth functions in 1D can be found in [10].

Each expert defines a hyper-plane m_j, and a window function w_j, which defines the operating region of the expert. The hyper-plane m_j describes a gradient, which indicates how the signal behaves around the center of the component (Eq. 4). The window function w_j gives weight to each sample, indicating the soft membership of that pixel to that component (Eq. 6). By modeling the correlation between sample location and amplitudes, our "local" SMoE components with "global" support can steer along edges and adopt regional signal intensity flow, similar to the "local" SKR [3].

Let us define in our special case $x^i \in \mathbb{R}^2$ as the locations of the $y^i \in \mathbb{R}$ amplitudes of an image. Regressing the model is equal to finding the most probable amplitude given a location $x = [x_1, x_2]$ through the conditional pdf $Y|X$ [8]:

$$p_Y(Y|X=x) = \sum_{j=1}^{K} w_j(x)\mathcal{N}(m_j(x), \sigma_j^2) \qquad (5)$$

with mixing weights

$$w_j(x) = \frac{\pi_j \mathcal{N}_j(\mu_{x_j}, R_{X_jX_j})}{\sum_{i=1}^{K} \pi_i \mathcal{N}_i(\mu_{x_i}, R_{X_iX_i})} \qquad (6)$$

Note that Eq. 6 corresponds to the *softmax* function frequently used in *artificial neural networks* and used to define the support of the model component. From Eq. 5 and 6 follows the *regression function* $m(x)$:

$$\hat{Y} = m(x) = \sum_{j=1}^{K} m_j(x) w_j(x) \qquad (7)$$

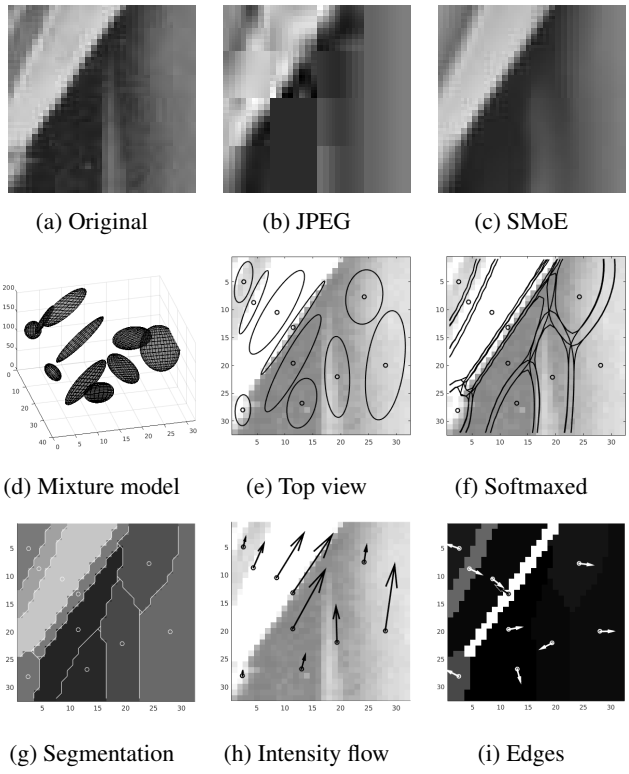
In general, any $(p+q)$-dimensional regression can be preformed this way. Thus we could e.g. include color as well the temporal domain for video sequences into the regression formula [5], or the angular dimensions for lightfield images. Note that elements of this regression have previously been used for the restoration of non-linear degraded images in [11].

2.2. Coding and Extracted Features

Fig. 1 depicts the excellent compression capability of the SMoE approach for coding a 32x32 pixel crop of *Lena* at 0.35 bit/sample in comparison to JPEG at same rate. The SMoE model parameters were quantized prior to reconstruction to arrive at the designated bit rate. For fair comparison, the bits required for the JPEG header were subtracted. It is apparent that especially the edges are reconstructed with impressive quality and sharpness. Fig. 1(d) shows the steering

of the 3-D ellipsoid Gaussian "cigars" components, which define the m_j "global" 2-D steering planes for regression. Fig. 1(e) illustrates steering of the ellipsoids projected onto the 2-D pixel domain. The respective softmax window functions dictate how the steering kernel planes are windowed. The windows overlap adaptively into adjacent image regions and enable either smoothness of the transition between regions or abrupt changes. The windows are of arbitrary shape and steer along edges. Thus, dominant edges are well reconstructed considering the low amount of components. Fine details and noise are eliminated which is the result of the very sparse representation with only 10 components. Note that the dominant gradient is very well approximated by only one component. Since the decoder arrives at a continuous, parametric regression equation, a super-resolution or downsampled version of the image with sharp edges at any scale is readily available. Also notice that the model yields a point-cloud, since for each component a feature vector is defined. Thus, the model can be coded using cloud-point coding algorithms. In addition, the coding format admits a graph representation [12].

Fig. 1 also illustrates that MPEG-7-like image descriptors are extracted directly from the decoded component matrix coefficients and centers. Since the SMoE approach follows a Bayesian interpretation, a segmentation of the image into K regions can be easily obtained by deriving the maximum posteriori probability of each image pixel from the window functions w_j. The segmentation boils down to determining for each pixel the most dominant component. In Fig 1, the center value of each expert defines the average gray value in the segment. The intensity flow (local orientation of a component) is the principle component of the decoded coefficients in $R_{X_jX_j}$, and the slope strength is defined as $|S_j|$, with S_j being the slope $R_{Y_jX_j}R^{-1}_{X_jX_j}$. The orientation of the local gradient is given by the decoded principle component of S_j.



(a) Original (b) JPEG (c) SMoE

(d) Mixture model (e) Top view (f) Softmaxed

(g) Segmentation (h) Intensity flow (i) Edges

Fig. 1: An example of the modeling with 10 components and reconstruction of a 32x32 pixel crop from *Lena*. The decoded coefficients provide MPEG-7-like functionalities.

3. CODING APPROACH

3.1. Modeling

The Expectation-Maximization (EM) algorithm is used to estimate the parameters $\Phi_j = (\pi_j, \mu_j, R_j)$ for every component j [13][14]. The optimization problem is unfortunately non-convex and converges to a local optimum [9]. In order to avoid local optima, a sparsification approach involving a split-and-merge algorithm was used to split undesired components, while merging two other [15]. In general it is important to arrive at few components in regions that are flat, but a larger amount in detailed areas. To ensure adequate granularity over the whole image, it is divided into blocks. Every block receives a different budget of components. Similar to [4], a 2D-DCT is performed and the spatial activity A_i for block i is calculated as the normalized squared sum of the first row and column of the AC coefficients. Note that the modeling is performed block-wise, but the reconstruction is global.

Given the average components per block K, and a spatial activity sensitivity parameter τ, the budget K_i for every block i is calculated as

$$K_i = K + \text{round}(\tau(A_i - E[A])K) \tag{8}$$

3.2. Quantization and entropy coding

The centers $\mu = [\mu_X, \mu_Y]$ are difference coded by defining a simple path that comprises every component in a greedy fashion. Start with the component j closest to $(0, 0)$. Find component $k, (k \neq j)$, so that $|\mu_j - \mu_k|$ is minimal. More advanced techniques from the point cloud coding field exist, but require signaling the path [16]. Finally, the difference coded centers are uniformly quantized.

At the decoder side, only $R_{X_jX_j}$ and $R_{X_jY_j}$ are needed for reconstruction of the images. $R_{X_jX_j}$ is coded as α_j, the angle of the eigenvector placed in the first quadrant, combined with $e_{j,1}, e_{j,2}$, the corresponding eigenvalues. The eigendecomposition allows for robust quantization. The 2-D covariances $R_{XY,j}$ are Laplacian distributed and quantized.

The population densities π_j are not coded, but estimated at the decoder side as the mean of the average population density $1/K$ and the relative size of the surface defined by the eigencomponents of $R_{X_jX_j}$:

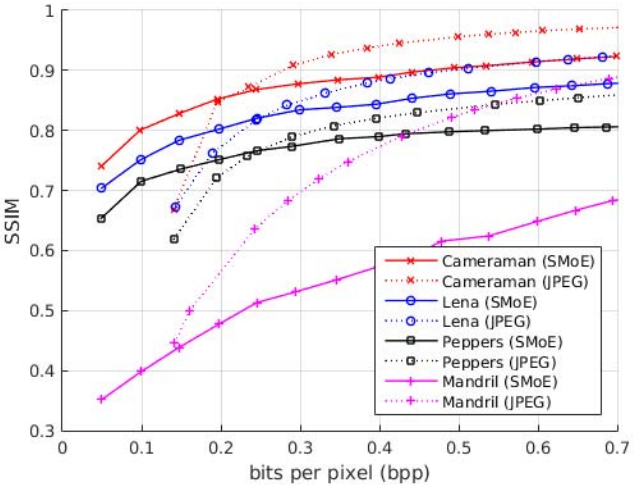


Fig. 2: Rate-distortion curves

$$\hat{\pi_j} = \left(\frac{1}{K} + \frac{e_{j_1} e_{j_2}}{\sum_{i=1}^{K} e_{i_1} e_{i_2}} \right) / 2 \qquad (9)$$

The quantized differences μ_j's, $R_{X_j Y_j}$'s are Laplacian distributed - a Laplacian adaptive arithmetic coder is employed as in [4]. Note that both the modeling and the coefficient quantization contribute to the approximation error.

3.3. Experiments

For coding experiments the EM algorithm was initialized by k-means++ with 7 repeats on the same data [17]. The bandwidths were initialized as 1e-3 for $R_{X_j X_j}$ and 0.15 for $R_{Y_j Y_j}$. Blocksizes B_i were 32, 64 or 128. For the allocation of bits, the following values were used: for centers μ_j, angles α_j, eigenvalues e_j and slope covariances $R_{X_j Y_j}$ ranges $[8, 11]$, $[4, 7]$, $[5, 6]$, and $[5, 9]$ bits per coefficient were tested respectively. The spatial activity sensitivity parameter τ_i had the range $[1.5, 1.75, 2, 2.25, 2.5]$. A maximum between 5 and 10 split-and-merge iterations were performed per block.

As SMoE is still an immature approach, we compare against JPEG to show that we achieve results that are at least in the same ballpark. We found that JPEG-2000 outperforms SMoE in general. Fig. 2 depicts the rate-distortion results for test images *Lena*, *Mandril*, *Cameraman* and *Peppers*, each 512x512 pels. A considerable compression gain is achieved for bitrates $<$ 0.25 bpp. It appears that the SMoE model with all steering parameters is, however, too elaborate to code fine details with high quality. Not surprisingly for image *Mandril* with its predominately high frequency content, negligible coding gain is achieved even at low rates. Fig. 3 shows the visual differences between JPEG and the "universal" SMoE approach for high and low rates. SMoE (27,1 dB) is able to reconstruct the dominant edges and smooth transition very well, while JPEG (25.0 dB) suffers from severe block artefacts at low rates (both at 0.141 bpp). For higher quality (0.45 bpp) our SMoE implementation achieves 30.1 dB and suffers from the lack of additional components to reconstruct the minor noise-like details, i.e. the model becomes too elaborate for coding each detail. JPEG achieves 32.7 dB at 0.45 bpp. Notice that in general the block-based JPEG coding approach results in block artefacts at low rates, while the SMoE strategy generates geometrical distortions. This is easily explained with reference to Fig. 1. More results, example files, and a MATLAB implementation can be found on: `http://users.elis.ugent.be/~rverhack`

Fig. 3: *Peppers* at 0.14 and 0.45 bpp: JPEG (left), GMR (right)

4. CONCLUSIONS AND FUTURE WORK

SMoE offers an attractive strategy for "universal" coding of images at low bit rates. At lower rates image quality is superior to JPEG and SMoE has the advantage that MPEG-7-like features are embedded in the bitstream. Although in order to be truly universal, texture information should be added by using conventional shape adaptive transforms, e.g. 3-D SA-DCT [18] or ideally embedded in our Bayesian framework. Further bit rate savings are possible by improving the modeling. Too many components are spent on flat regions. Also, the log-likelihood criteria used in the EM algorithm may not guide the model towards the intended goal. To overcome this, posterior constraints should be added to steer the algorithm [19]. Finally, an efficient more flexible non-blockbased modeling approach could be developed.

5. REFERENCES

[1] T. Sikora, "Trends and Perspectives in Image and Video Coding," *Proceedings of the IEEE*, vol. 93, no. 1, pp. 6–17, jan 2005.

[2] T. Sikora, "The MPEG-7 Visual Standard for Content Description – An Overview," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 11, no. 6, pp. 696–702, jun 2001.

[3] H. Takeda, S. Farsiu, and P. Milanfar, "Kernel Regression for Image Processing and Reconstruction," *Image Processing, IEEE Transactions on*, vol. 16, no. 2, pp. 349–366, 2007.

[4] R. Verhack, A. Krutz, P. Lambert, R. Van de Walle, and T. Sikora, "Lossy Image Coding in the Pixel Domain using a Sparse Steering Kernel Synthesis Approach," in *2014 IEEE International Conference on Image Processing (ICIP)*. oct 2014, pp. 4807–4811, IEEE.

[5] L. Lange, R. Verhack, and T. Sikora, "Sparse Steered Mixture-of-Experts Regression for Universal Video Coding," in *submitted to 2016 Picture Coding Symposium (PCS)*, 2016.

[6] A. Smola and B. Schölkopf, "A Tutorial on Support Vector Regression," *Statistics and computing*, vol. 14, no. 3, pp. 199–222, 2004.

[7] R. Jacobs, M. Jordan, S. Nowlan, and G. Hinton, "Adaptive Mixtures of Local Experts," *Neural Computation*, vol. 3, no. 1, pp. 79–87, feb 1991.

[8] H. Sung, *Gaussian Mixture Regression and Classification*, Ph.D. thesis, Rice University, 2004.

[9] T. Moon, "The Expectation-Maximization Algorithm," *IEEE Signal Processing Magazine*, vol. 13, no. 6, pp. 47–60, 1996.

[10] P. Prandoni and M. Vetterli, "Approximation and Compression of Piecewise Smooth Functions," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 357, no. 1760, pp. 2573–2591, sep 1999.

[11] I. Cha and S. Kassam, "RBFN Restoration of Nonlinearly Degraded Images," *IEEE Transactions on Image Processing*, vol. 5, no. 6, pp. 964–975, jun 1996.

[12] D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst, "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains," *IEEE Signal Processing Magazine*, vol. 30, no. 3, pp. 83–98, may 2013.

[13] A. Dempster, N. Laird, and D. Rubin, "Maximum Likelihood from Incomplete Data via the EM Algorithm," *Journal of the royal statistical society. Series B (methodological)*, pp. 1–38, 1977.

[14] M. Jordan and L. Xu, "Convergence Results for the EM Approach to Mixtures of Experts Architectures," *Neural Networks*, vol. 8, no. 9, pp. 1409–1431, jan 1995.

[15] Z. Zhang, C. Chen, J. Sun, and K. Luk Chan, "EM algorithms for Gaussian Mixtures with Split-and-Merge Operation," *Pattern Recognition*, vol. 36, no. 9, pp. 1973–1983, sep 2003.

[16] S. Gumhold, Z. Kami, M. Isenburg, and H.-P. Seidel, "Predictive Point-Cloud Compression," in *ACM SIGGRAPH 2005 Sketches on - SIGGRAPH '05*, New York, USA, 2005, p. 137, ACM Press.

[17] D. Arthur and S. Vassilvitskii, "K-means++: The Advantages of Careful Seeding," in *SODA '07 Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, 2007, pp. 1027–1035.

[18] T. Sikora and B. Makai, "Shape-adaptive DCT for generic coding of video," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 5, no. 1, pp. 59–62, 1995.

[19] K. Ganchev, J. Graça, J. Gillenwater, and B. Taskar, "Posterior Regularization for Structured Latent Variable Models," *The Journal of Machine Learning Research*, vol. 11, pp. 2001–2049, 2010.